Chapter 13

Digital Signature

Chapter 13
Objectives

- To define a digital signature
- To define security services provided by a digital signature
- To define attacks on digital signatures
- To discuss some digital signature schemes, including RSA, ElGamal,
- Schnorr, DSS, and elliptic curve
- To describe some applications of digital signatures

13-1 COMPARISON

Let us begin by looking at the differences between conventional signatures and digital signatures.

Topics discussed in this section:

13.1.1 Inclusion

A conventional signature is included in the document; it is part of the document. But when we sign a document digitally, we send the signature as a separate document.

13.1.2 Verification Method

For a conventional signature, when the recipient receives a document, she compares the signature on the document with the signature on file. For a digital signature, the recipient receives the message and the signature. The recipient needs to apply a verification technique to the combination of the message and the signature to verify the authenticity.

13.1.3 Relationship

For a conventional signature, there is normally a one-to-many relationship between a signature and documents. For a digital signature, there is a one-to-one relationship between a signature and a message.

13.1.4 Duplicity

In conventional signature, a copy of the signed document can be distinguished from the original one on file. In digital signature, there is no such distinction unless there is a factor of time on the document.

13-2 PROCESS

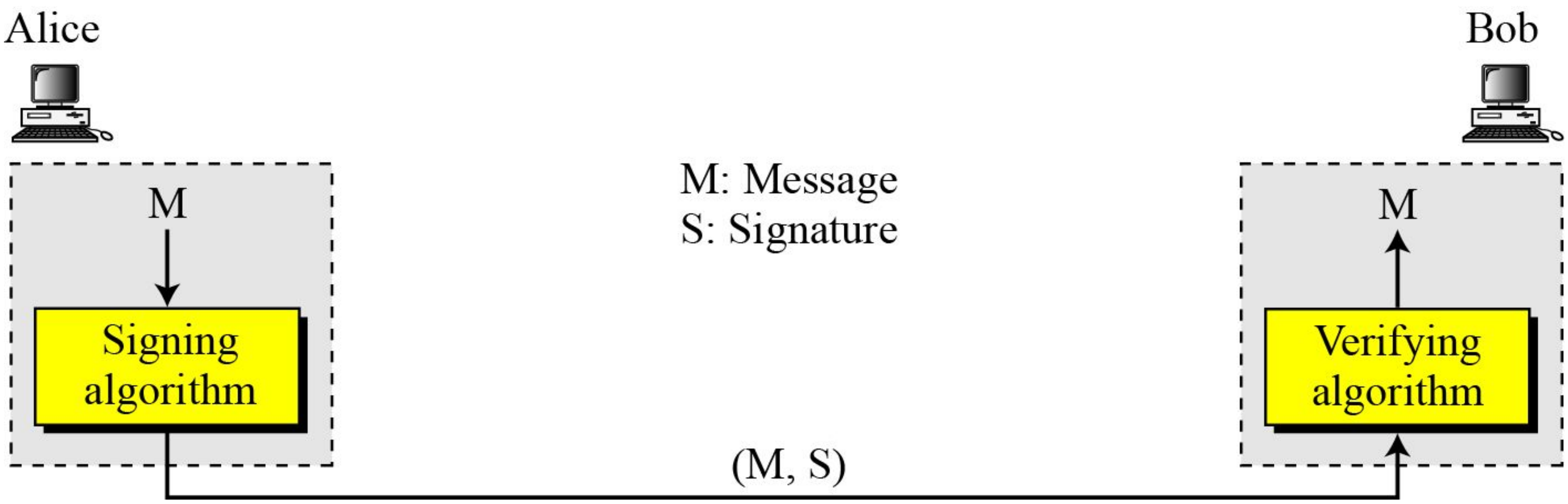
Figure 13.1 shows the digital signature process. The sender uses a signing algorithm to sign the message. The message and the signature are sent to the receiver. The receiver receives the message and the signature and applies the verifying algorithm to the combination. If the result is true, the message is accepted; otherwise, it is rejected.

Topics discussed in this section:

13.2.1 Need for Keys
13.2.2 Signing the Digest

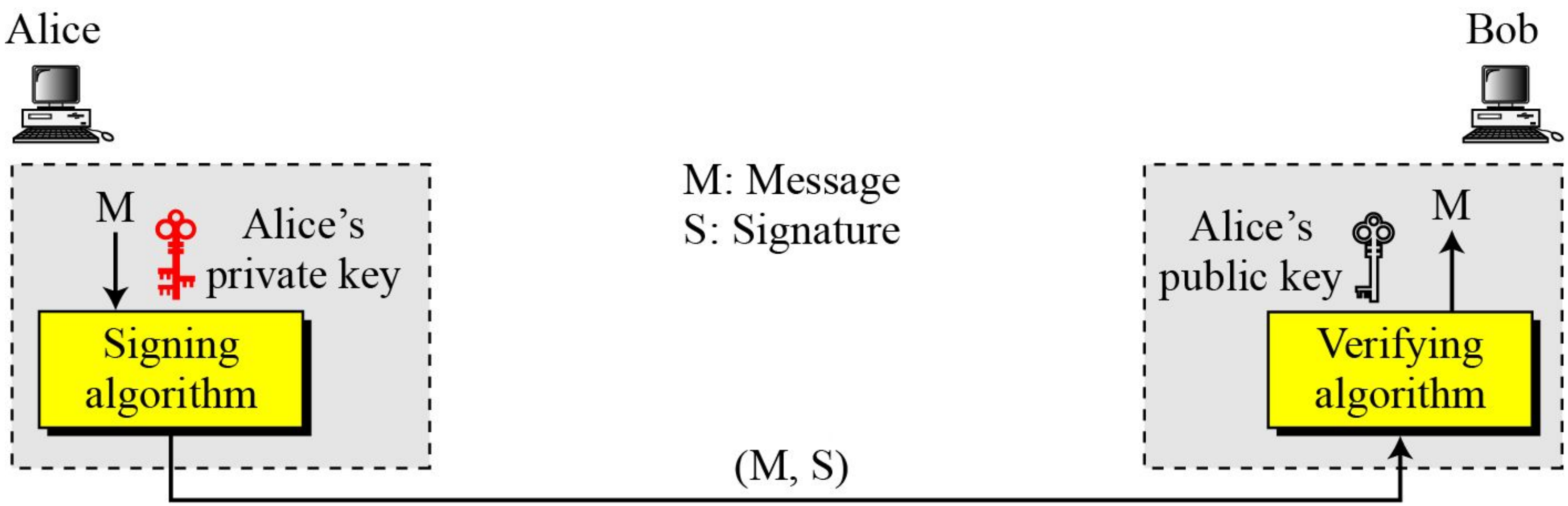
13-2 Continued

Figure 13.1 ***Digital signature process***



13.2.1 Need for Keys

Figure 13.2 *Adding key to the digital signature process*



Note

A digital signature needs a public-key system. The signer signs with her private key; the verifier verifies with the signer's public key.

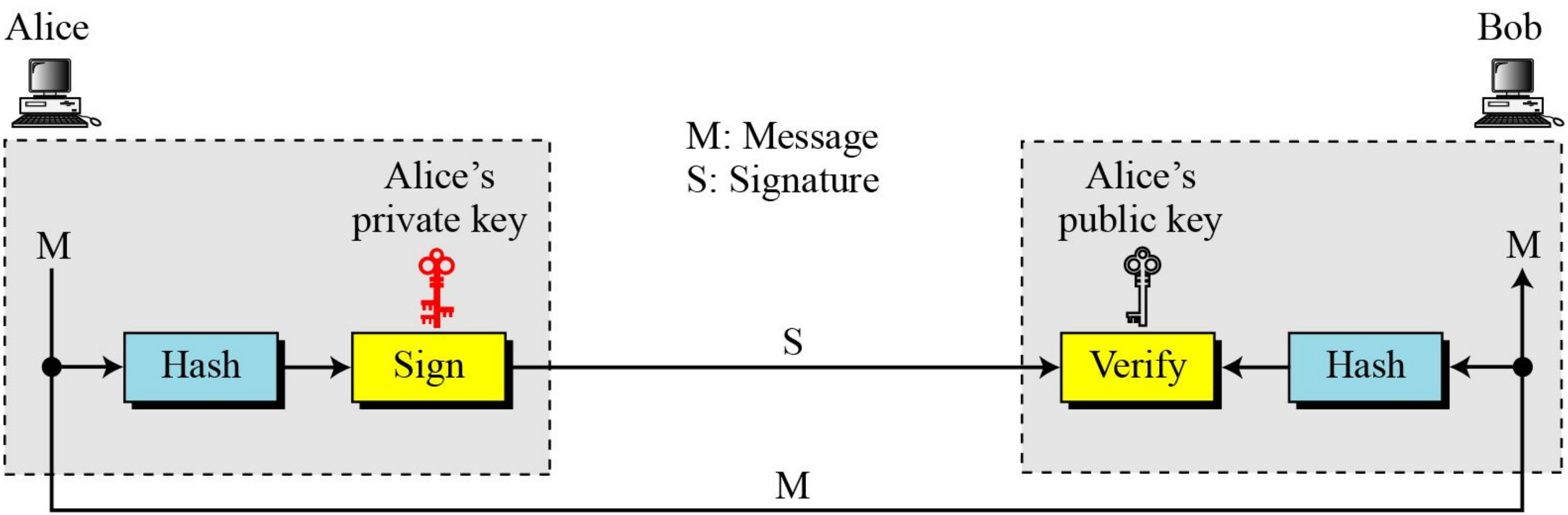
13.2.1 Continued

Note

A cryptosystem uses the private and public keys of the receiver: a digital signature uses the private and public keys of the sender.

13.2.2 Signing the Digest

Figure 13.3 ***Signing the digest***



13-3 SERVICES

We discussed several security services in Chapter 1 including message confidentiality, message authentication, message integrity, and nonrepudiation. A digital signature can directly provide the last three; for message confidentiality we still need encryption/decryption.

Topics discussed in this section:

13.3.1 Message Authentication
13.3.2 Message Integrity
13.3.3 Nonrepudiation
13.3.4 Confidentiality

13.3.1 Message Authentication

A secure digital signature scheme, like a secure conventional signature can provide message authentication.

Note

A digital signature provides message authentication.

13.3.2 Message Integrity

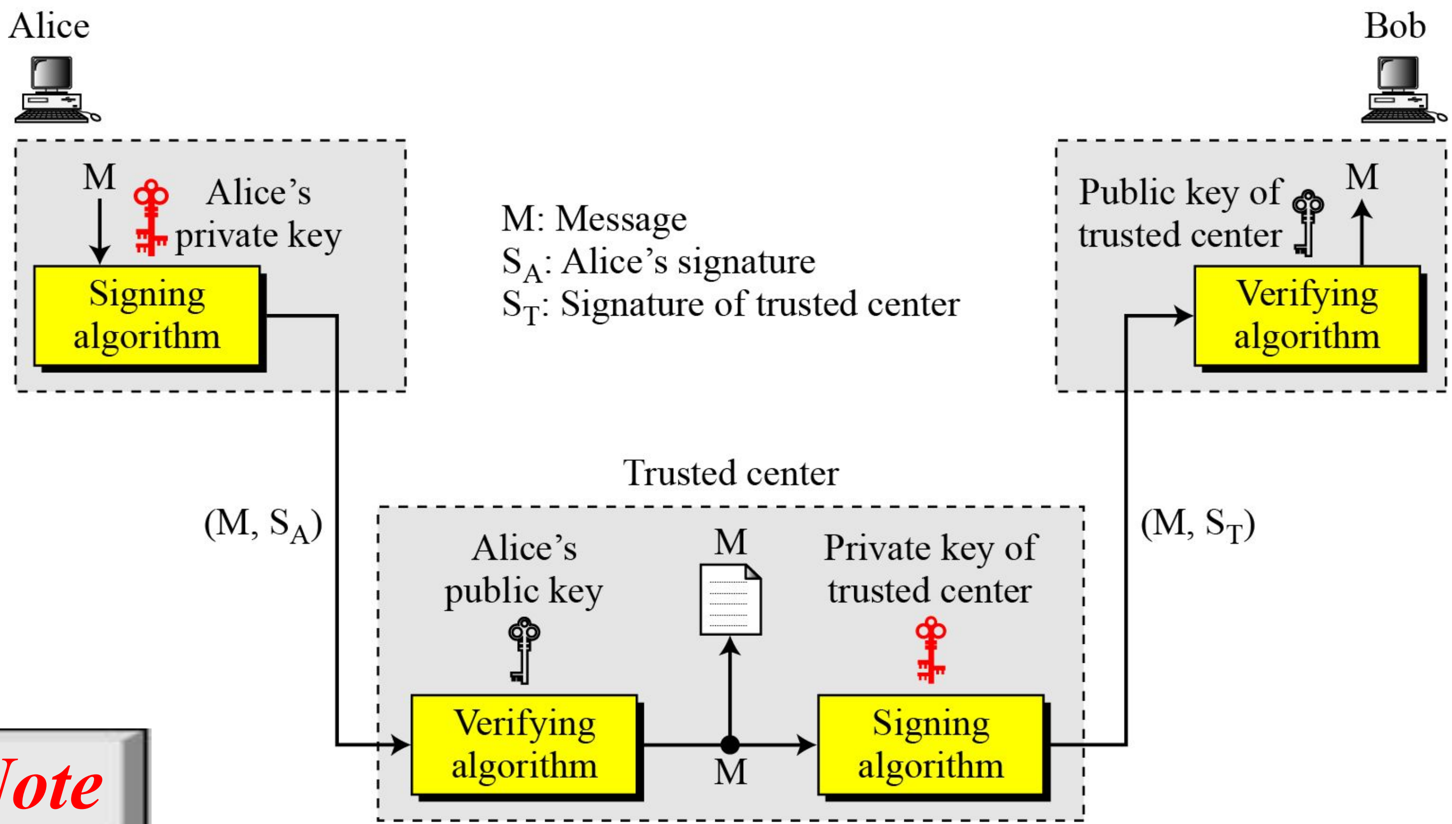
The integrity of the message is preserved even if we sign the whole message because we cannot get the same signature if the message is changed.

Note

A digital signature provides message integrity.

13.3.3 Nonrepudiation

Figure 13.4 *Using a trusted center for nonrepudiation*

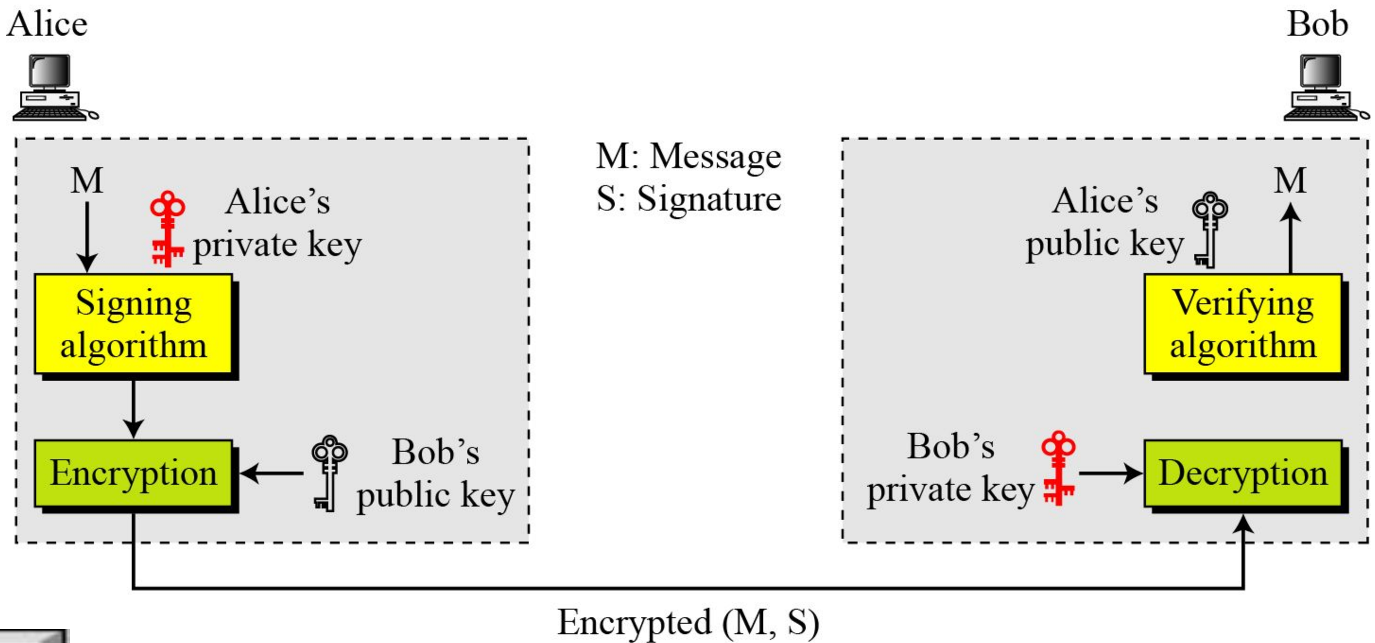


Note

Nonrepudiation can be provided using a trusted party.

13.3.4 Confidentiality

Figure 13.5 ***Adding confidentiality to a digital signature scheme***



Note

A digital signature does not provide privacy. If there is a need for privacy, another layer of encryption/decryption must be applied.

13-4 ATTACKS ON DIGITAL SIGNATURE

This section describes some attacks on digital signatures and defines the types of forgery.

Topics discussed in this section:

13.4.1 Attack Types
13.4.2 Forgery Types

13.4.1 Attack Types

Key-Only Attack

Known-Message Attack

Chosen-Message Attack

13.4.2 Forgery Types

Existential Forgery

Selective Forgery

13-5 DIGITAL SIGNATURE SCHEMES

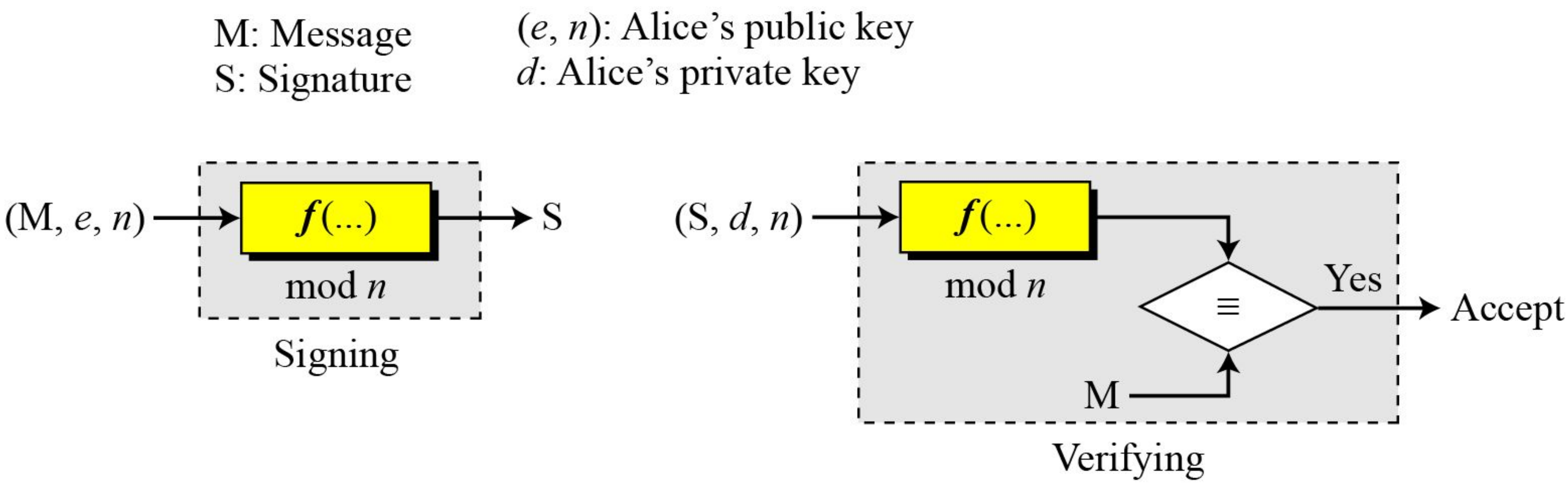
Several digital signature schemes have evolved during the last few decades. Some of them have been implemented.

Topics discussed in this section:

13.5.1 RSA Digital Signature Scheme
13.5.2 ElGamal Digital Signature Scheme
13.5.3 Schnorr Digital Signature Scheme
13.5.4 Digital Signature Standard (DSS)
13.5.5 Elliptic Curve Digital Signature Scheme

13.5.1 RSA Digital Signature Scheme

Figure 13.6 ***General idea behind the RSA digital signature scheme***



13.5.1 Continued

Key Generation

Key generation in the RSA digital signature scheme is exactly the same as key generation in the RSA

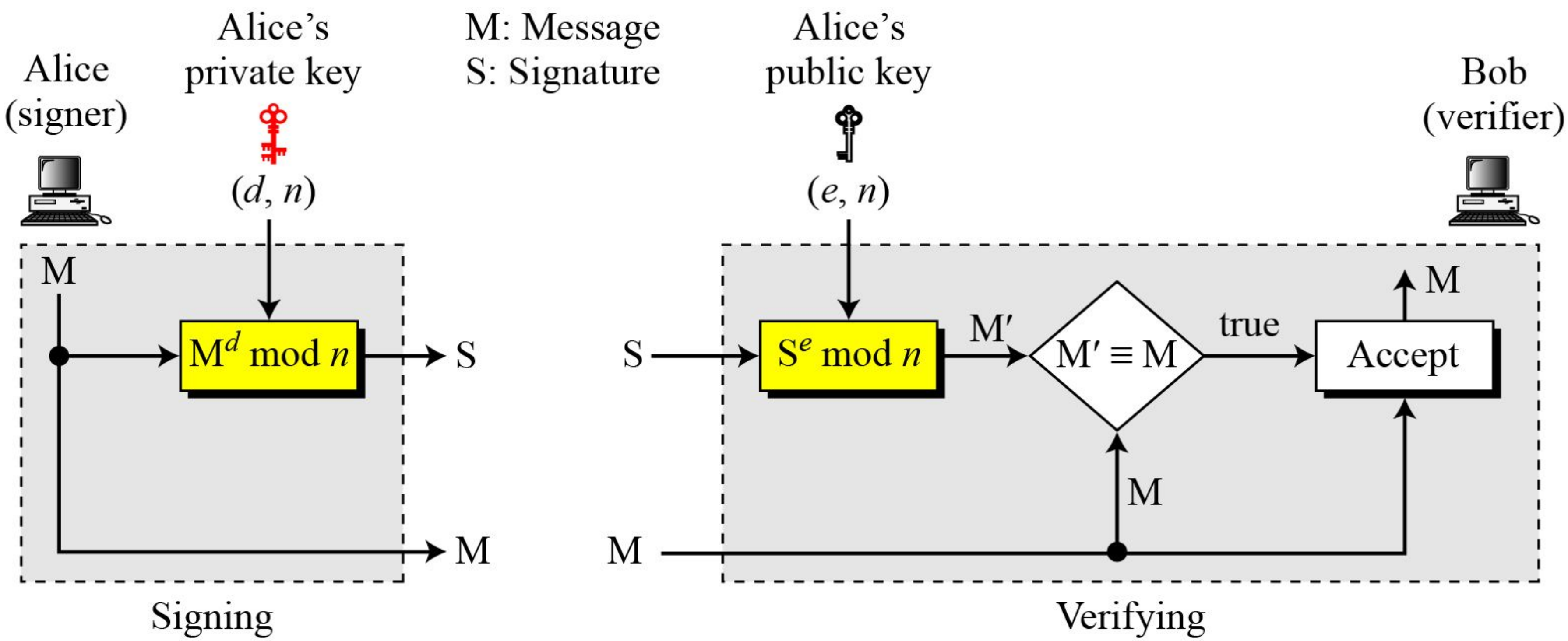
Note

In the RSA digital signature scheme, *d* is private; *e* and *n* are public.

13.5.1 Continued

Signing and Verifying

Figure 13.7 ***RSA digital signature scheme***



Example 13.1

As a trivial example, suppose that Alice chooses *p* = 823 and *q* = 953, and calculates *n* = 784319. The value of φ(n) is 782544. Now she chooses *e* = 313 and calculates *d* = 160009. At this point key generation is complete. Now imagine that Alice wants to send a message with the value of M = 19070 to Bob. She uses her private exponent, 160009, to sign the message:

$$\text{M: } 19070 \quad \rightarrow \quad \text{S} = (19070^{160009}) \bmod 784319 = 210625 \bmod 784319$$

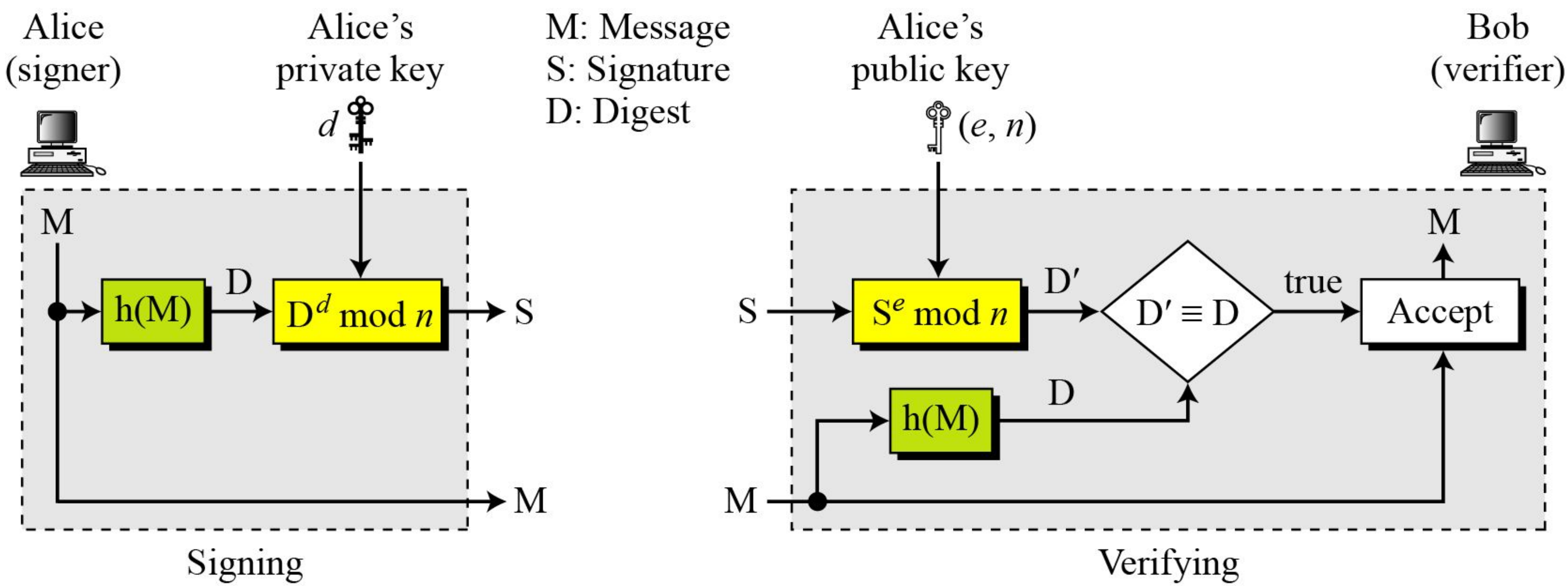
Alice sends the message and the signature to Bob. Bob receives the message and the signature. He calculates

$$\text{M}' = 210625^{313} \bmod 784319 = 19070 \bmod 784319 \quad \rightarrow \quad \text{M} \equiv \text{M}' \bmod n$$

Bob accepts the message because he has verified Alice's signature.

RSA Signature on the Message Digest

Figure 13.8 *The RSA signature on the message digest*



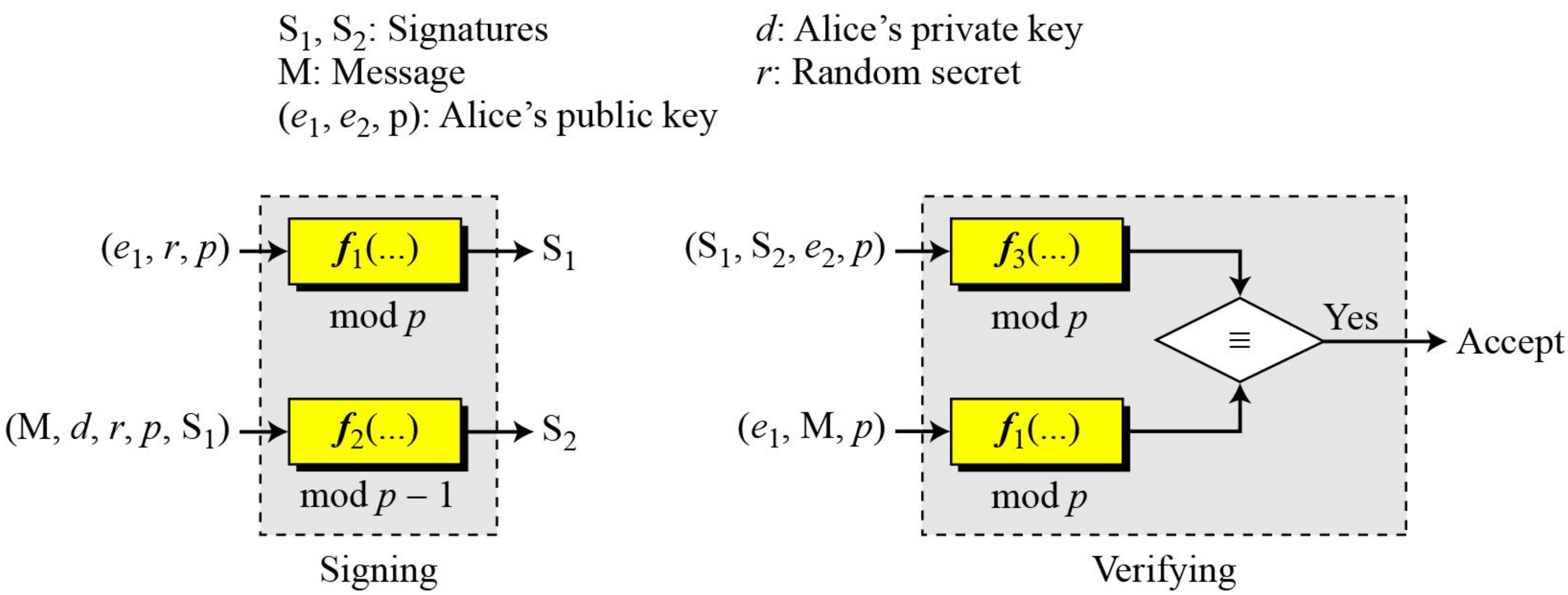
13.5.1 Continued

Note

When the digest is signed instead of the message itself, the susceptibility of the RSA digital signature scheme depends on the strength of the hash algorithm.

13.5.2 ElGamal Digital Signature Scheme

Figure 13.9 ***General idea behind the ElGamal digital signature scheme***



13.5.2 Continued

Key Generation

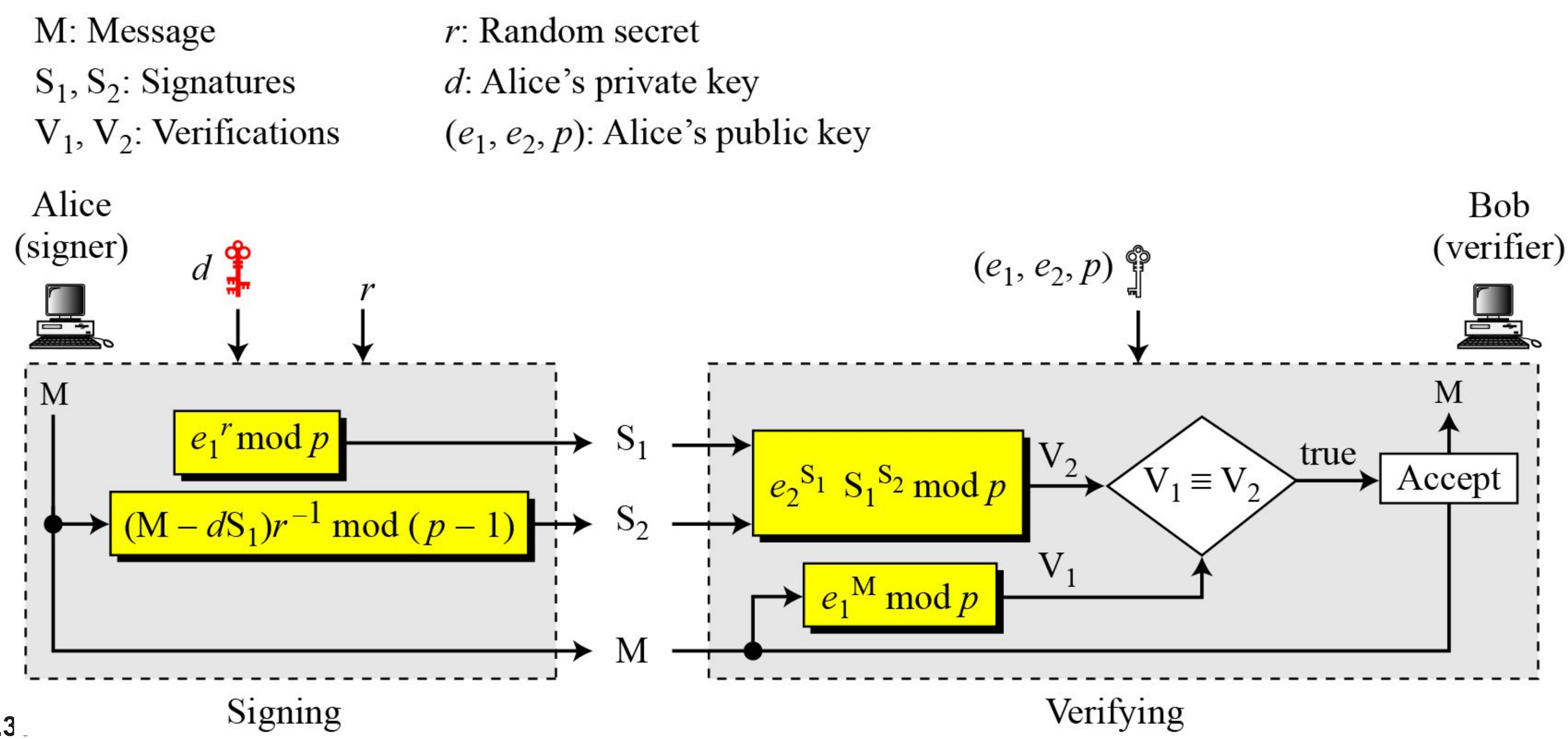
The key generation procedure here is exactly the same as the one used in the cryptosystem.

Note

In ElGamal digital signature scheme, (e_1, e_2, p) is Alice's public key; d is her private key.

Verifying and Signing

Figure 13.10 *ElGamal digital signature scheme*



Example 13.2

Here is a trivial example. Alice chooses p = 3119, e_1 = 2, $_d$ = 127 and calculates $e_2 = 2^{127} \bmod 3119 = 1702$. She also chooses r to be 307. She announces e*1*, e*2*, and p publicly; she keeps d secret. The following shows how Alice can sign a message.

$$M = 320$$
$$S_1 = e_1{}^r = 2^{307} = 2083 \bmod 3119$$
$$S_2 = (M - d \times S_1) \times r^{-1} = (320 - 127 \times 2083) \times 307^{-1} = 2105 \bmod 3118$$

Alice sends M, S_1, and S_2 to Bob. Bob uses the public key to calculate V_1 and V_2.

$$V_1 = e_1{}^{M} = 2^{320} = 3006 \bmod 3119$$
$$V_2 = d^{S_1} \times S_1{}^{S_2} = 1702^{2083} \times 2083^{2105} = 3006 \bmod 3119$$

Example 13.3

Now imagine that Alice wants to send another message, M = 3000, to Ted. She chooses a new *r*, 107. Alice sends M, S_1, and S_2 to Ted. Ted uses the public keys to calculate V_1 and V_2.

$$M = 3000$$
$$S_1 = e_1{}^r = 2^{107} = 2732 \bmod 3119$$
$$S_2 = (M - d \times S_1)\ r^{-1} = (3000 - 127 \times 2083) \times 107^{-1} = 2526 \bmod 3118$$

$$V_1 = e_1{}^{M} = 2^{3000} = 704 \bmod 3119$$
$$V_2 = d^{\,S_1} \times S_1{}^{S} = 1702^{2732} \times 2083^{2526} = 704 \bmod 3119$$